A General Framework for Modeling and Online Optimization of Stochastic Hybrid Systems ⋆

A. Kebarighotbi * **C. G. Cassandras** *

* *Division of Systems Engineering, Center for Information and Systems Engineering, Boston University, Brookline, MA 02446*
alik@bu.edu,cgc@bu.edu

Abstract: We extend the definition of a Stochastic Hybrid Automaton (SHA) to overcome limitations that make it difficult to use for on-line control. Since guard sets do not specify the exact event causing a transition, we introduce a clock structure (borrowed from timed automata), timer states, and guard functions that disambiguate how transitions occur. In the modified SHA, we formally show that every transition is associated with an explicit element of an underlying event set. This also makes it possible to uniformly treat all events observed on a sample path of a stochastic hybrid system and generalize the performance sensitivity estimators derived through Infinitesimal Perturbation Analysis (IPA). We eliminate the need for a case-by-case treatment of different event types and provide a unified set of matrix IPA equations. We illustrate our approach by revisiting an optimization problem for single node finite-capacity stochastic flow systems to obtain performance sensitivity estimates in this new setting.

Keywords: Hybrid Systems, Discrete-Event Systems, Perturbation Analysis

1. INTRODUCTION

A Stochastic Hybrid System (SHS) consists of both time-driven and event-driven components. Its stochastic features may include random transition times and external stochastic inputs or disturbances. The modeling and optimization of these systems is quite challenging and many models have been proposed, some capturing randomness through probabilistic resets when reset functions are distributions, through spontaneous transitions at random times [Bujorianu and Lygeros, 2006],[Hespanha, 2004], Stochastic Differential Equations (SDE) [Ghosh et al., 1993],[Ghosh et al., 1997], or using Stochastic Flow Models (SFM) [Cassandras et al., 2002] with the aim of describing stochastic continuous dynamics.

Optimizing the performance of SHS poses additional challenges and most approaches rely on approximations and/or using computationally taxing methods. For example, [Bujorianu and Lygeros, 2004] and [Koutsoukos, 2005] resort to dynamic programming techniques. The inherent computational complexity of these approaches makes them unsuitable for on-line optimization. However, in the case of parametric optimization, application of *Infinitesimal Perturbation Analysis* (IPA) [Cassandras et al., 2002] to SHS has been very successful in on-line applications. Using IPA, one generates sensitivity estimates of a performance objective with respect to a control parameter vector based on readily available data observed from a single sample path of the SHS. Along this line, SFMs provide the most common framework for applying IPA in the SHS setting and have their root in making abstractions of complex *Discrete Event Systems* (DES), where the event rates are treated as stochastic processes of arbitrary generality except for mild technical assumptions. A fundamental property of IPA is that the derivative estimates obtained are independent of the probability laws of the stochastic processes involved. Thus, they can be easily obtained and, unlike most other techniques, they can be implemented in on-line algorithms.

In this paper, we aim at extending *Stochastic Hybrid Automata* (SHA) [Cassandras and Lygeros, 2006] and create a framework within which IPA becomes straightforward and applicable to arbitrary SHS. A SHA, specifies discrete states (or *modes*) where the state $\mathbf{x}$ evolves according to a continuous vector field until an event triggers a mode transition. The transitions are described by *guards* and *invariants* as well as clock structures [Bujorianu and Lygeros, 2006] borrowed from *Stochastic Timed Automata* (STA) [Glynn, 1989],[Cassandras and Lafortune, 2006]. When $\mathbf{x}$ reaches a guard set, a transition becomes *enabled* but not triggered. On the other hand, if $\mathbf{x}$ exits the boundary defined by the invariant set in a mode or if a spontaneous event occurs, the transition must trigger. This setting has the following drawbacks: (i) The clock structure (normally part of the system input) is not incorporated in the definition of guards and invariant conditions. As a result, spontaneous transitions have to be treated differently. (ii) The guard set does not specify the exact event causing a transition and we cannot, therefore, differentiate between an event whose occurrence time may depend on some control parameter $\boldsymbol{\theta}$ and another that does not. This is a crucial point in IPA, as it directly affects how perfor-

⋆ The authors' work is supported in part by NSF under Grant EFRI-0735974, by AFOSR under grant FA9550-09-1-0095, by DOE under grant DE-FG52-06NA27490, by ONR under grant N00014-09-1-1051, and by ARO under grant W911NF-11-1-0227.

mance derivative estimates evolve in time. As described in [Cassandras et al., 2009], when applied to SFMs, IPA uses a classification of events into different types (exogenous, endogenous, and induced) to extract this information.

Here, we seek an enriched model which explicitly specifies the event triggering a transition and, at the same time, creates a unified treatment for all events, i.e., all IPA equations are common regardless of event type. We achieve this by introducing state variables representing *timers* and treating a clock structure as an input to the system with the mode invariants generally dependent on both. We formalize the definition of an event by associating it to a *guard function* replacing the notion of the guard set. This removes the ambiguity caused by an enabled, but not triggered, event, as well as the need for treating spontaneous transitions differently. A byproduct of this unified treatment is the development of a matrix notation for the IPA equations, which makes the treatment of complex systems with multiple states and events a straightforward application of these equations. We verify this process by applying it to a single node queueing system previously solved using the SFM framework [Cassandras et al., 2002].

The paper is organized as follows: In Section 2, we present the general SHA which includes all the features previously handled by SFMs. Utilizing the resulting model, in Section 3 we develop a matrix notation for IPA which simplifies the derivation of sensitivity estimators in Section 4. We verify our results in Section 5 by applying the proposed technique to a single node finite capacity buffer system. We conclude with Section 6.

2. GENERAL OPTIMIZATION MODEL

Let us consider a *Stochastic Hybrid Automaton* (SHA) as defined in [Cassandras and Lafortune, 2006] with only slight modifications and parameterized by the vector $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_{N_\theta}) \in \Theta$ as

$$G = (\mathcal{Q}, \mathcal{X}, \mathcal{U}, \Theta, \mathcal{E}, \mathbf{f}, \phi, Inv, guard, \mathbf{r}, (q_0, \mathbf{x}_0))$$

where

- $\mathcal{Q} \subset \mathbb{Z}^+$ is the countable set of discrete states or modes;
- $\mathcal{X}(\boldsymbol{\theta}) \subset \mathbb{R}^{N_x}$ is the admissible continuous state space for any $\boldsymbol{\theta} \in \Theta$;
- $\mathcal{U}(\boldsymbol{\theta}) \subset \mathbb{R}^{N_u}$ is the set of inputs (possibly disturbances or clock variables) for any $\boldsymbol{\theta} \in \Theta$;
- $\Theta \subset \mathbb{R}^{N_\theta}$ is the set of admissible control parameters;
- $\mathcal{E}$ is a countable event set $\mathcal{E} = \{E_i, i = 1, 2, \ldots, N_e\}$;
- $\mathbf{f}$ is a continuous vector field, $\mathbf{f} : \mathcal{Q} \times \mathcal{X}(\boldsymbol{\theta}) \times \mathcal{U}(\boldsymbol{\theta}) \times \Theta \mapsto \mathcal{X}(\boldsymbol{\theta})$;
- ϕ is a discrete transition function $\phi : \mathcal{Q} \times \mathcal{X}(\boldsymbol{\theta}) \times \mathcal{U}(\boldsymbol{\theta}) \times \mathcal{E} \mapsto \mathcal{Q}$;
- Inv is a set defining an invariant condition such that $Inv \subseteq \mathcal{Q} \times \mathcal{X}(\boldsymbol{\theta}) \times \mathcal{U}(\boldsymbol{\theta}) \times \Theta$;
- $guard$ is a set defining a guard condition, $guard \subseteq \mathcal{Q} \times \mathcal{Q} \times \mathcal{X}(\boldsymbol{\theta}) \times \mathcal{U}(\boldsymbol{\theta}) \times \Theta$;
- $\mathbf{r}$ is a reset function, $\mathbf{r} : \mathcal{Q} \times \mathcal{Q} \times \mathcal{E} \times \mathcal{X}(\boldsymbol{\theta}) \times \mathcal{U}(\boldsymbol{\theta}) \times \Theta \mapsto \mathcal{X}(\boldsymbol{\theta})$;
- $(q_0, \mathbf{x}_0)$ is the initial state.

Note that the input $u \in \mathcal{U}(\boldsymbol{\theta})$ can be a vector of random processes which are all defined on a common probability space $(\Omega, \mathcal{F}, P)$. Also, observe that invariants and guards are sets which do not specify the events (hence, precise times) causing violation or adherence to their set conditions. Thus, we cannot differentiate between a transition that depends on $\boldsymbol{\theta}$ and one that does not. This prevents us from properly estimating the effect of $\boldsymbol{\theta}$ on the system behavior. In particular, if $(\mathbf{x}, \mathbf{u}) \in guard(q, q')$ for some $q, q' \in \mathcal{Q}$, a transition to $q' \in \mathcal{Q}$ can occur either through a policy that uniquely specifies some $(\mathbf{x}, \mathbf{u}, \boldsymbol{\theta})$ causing the transition or at some random time while $(\mathbf{x}, \mathbf{u}, \boldsymbol{\theta})$ remains in the guard set. This is one of the issues we focus on in what follows.

We allow the parameter vector $\boldsymbol{\theta}$ to affect the system not only through the vector field, reset conditions, guards, and invariants, but also its structure through $\mathcal{X}(\boldsymbol{\theta})$ and $\mathcal{U}(\boldsymbol{\theta})$, e.g., the parameters can appear in the state and input constraints. We remove $\boldsymbol{\theta}$ from the arguments whenever it does not cause any confusion and simplifies notation. Defining $\mathbf{x}(t, \boldsymbol{\theta})$ and $\mathbf{u}(t, \boldsymbol{\theta})$ as the state and input vectors, we introduce the following assumptions:

Assumption 1. With probability 1, for any $\mathbf{x} \in \mathcal{X}, q \in \mathcal{Q}, \mathbf{u} \in \mathcal{U}$, and $\boldsymbol{\theta} \in \Theta$, $\|\mathbf{f}(q, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta})\|_\infty < \infty$ where $\|\cdot\|_\infty$ is the L_∞ norm.

Assumption 2. With probability 1, no two events can occur at the same time unless one causes the occurrence of the other.

Assumption 1 ensures that $\mathbf{x}(t, \boldsymbol{\theta})$ remains smooth inside a mode as is embedded in the definition of the SHA above. Assumption 2 rules out the pathological case of having two independent events happening at the same time.

Borrowing the concept of clock structure from *Stochastic Timed Automata* as defined in [Cassandras and Lafortune, 2006], we associate an event $E_i \in \mathcal{E}$ with a a sequence $\{V_{i,1}(\boldsymbol{\theta}), V_{i,2}(\boldsymbol{\theta}), \ldots\}$ where $V_{i,n}(\boldsymbol{\theta})$ is the nth (generally random) lifetime of E_i, i.e., if this event becomes feasible for the nth time at some time t, then its next occurrence is at $t + V_{i,n}(\boldsymbol{\theta})$. Obviously, not all events in $\mathcal{E}$ are defined to occur in this fashion, but if they are, we define a *timer* as a state variable, say y_i, so that it is initialized to $y_i(t) = V_{i,n}(\boldsymbol{\theta})$ if E_i becomes feasible for the nth time at time t. Subsequently, the timer dynamics are given by $\dot{y}_i = -1$ until the timer runs off, i.e., $y_i(t + V_{i,n}(\boldsymbol{\theta})) = 0$. Figure 1 shows an example of a timer state as it evolves according to the supplied event lifetimes. We assume that

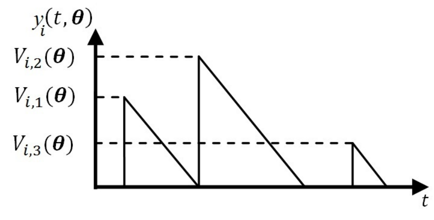

Fig. 1. A timer realization based on a given clock sequence $\{V_{i,n}\}$

$V_{i,n}(\boldsymbol{\theta})$ is differentiable with respect to $\boldsymbol{\theta}$ for all n.

The concept of event. In the SHA as defined above, $\mathcal{E}$ is simply a set of labels. We provide more structure to an event by assigning to each $E_i \in \mathcal{E}$, a *guard function* $g_i : \mathbb{R}^+ \times \mathcal{X} \times \mathcal{U} \times \Theta \mapsto \mathbb{R}$ which is not null (i.e. $g_i \not\equiv 0$) and is assumed to be differentiable almost everywhere on its domain. We are interested in a sample path of a SHA G on some interval $[0, T]$ where we let $\tau_k(\boldsymbol{\theta})$ be the time

when the kth transition fires and set $0 = \tau_0 \leq \tau_1(\boldsymbol{\theta}) \leq \ldots \leq \tau_K(\boldsymbol{\theta}) \leq \tau_{K+1} = T$. We then define an event E_i as occurring at time $\tau_k(\boldsymbol{\theta})$ if

$$\tau_k = \inf\{t \geq \tau_{k-1} : g_i(t, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}) = 0\}$$

that is, the event satisfies the condition $g_i(t, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}) = 0$ which was not being satisfied over $(\tau_{k-1}(\boldsymbol{\theta}), \tau_k(\boldsymbol{\theta}))$. The following theorem shows that using guard functions, we can associate every transition with an event occurring at the transition time.

Theorem 1. For every STA G with $\mathcal{E}$, $\mathcal{X}$ and $\mathcal{Q}$, there exists another STA $\tilde{G}$ with event set $\tilde{\mathcal{E}}$, continuous state space $\tilde{\mathcal{X}}$ and discrete state space $\tilde{\mathcal{Q}}$ such that $\tilde{\mathcal{Q}} = \mathcal{Q}$ and every transition (q, q') in G is associated with an event $e \in \tilde{\mathcal{E}}$ at the transition time.

Proof: A transition (q, q') is dictated by the transition function $\phi(q, \mathbf{x}, \mathbf{u}, e)$ such that $\phi(q, \mathbf{x}, \mathbf{u}, e) = q'$ for some $\mathbf{x} \in \mathcal{X}(\boldsymbol{\theta})$, $\mathbf{u} \in \mathcal{U}(\boldsymbol{\theta})$, $e \in \mathcal{E}$. If $\phi(q, \mathbf{x}, \mathbf{u}, E_i) = \phi(q, E_i) = q'$ for some $E_i \in \mathcal{E}$, i.e., the transition (q, q') is independent of $\mathbf{x} \in \mathcal{X}(\boldsymbol{\theta})$, $\mathbf{u} \in \mathcal{U}(\boldsymbol{\theta})$, the proof is complete. In this case, we can always augment $\mathbf{x} \in \mathcal{X}$ to $\tilde{\mathbf{x}} = (\mathbf{x}, y_i) \in \tilde{\mathcal{X}}$ where y_i is a timer state variable capturing lifetimes of event $E_i \in \mathcal{E}$ and associate E_i with guard function $g_i(t, \tilde{\mathbf{x}}, \mathbf{u}, \boldsymbol{\theta}) = y_i$. If $\phi(q, \mathbf{x}, \mathbf{u}, e) = \phi(q, \mathbf{x}, \mathbf{u}) = q'$, i.e., the transition (q, q') depends on $(\mathbf{x}, \mathbf{u}, \boldsymbol{\theta})$, then it is either a result of violating $Inv(q)$ or it occurs while $(\mathbf{x}, \mathbf{u}, \boldsymbol{\theta}) \in guard(q, q')$. In the former case, we can define some E_i such that $g_i(t, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}) = 0$ is the condition that determines the occurrence time of E_i. This is because $Inv(q)$ can be violated in two ways: (a) directly, due to an occurrence of E_i meaning $(\mathbf{x}, \mathbf{u})$ is on the boundary of $Inv(q)$ at the transition time; (b) indirectly, due to a reset condition which is the result of a previous transition $\phi(q', \mathbf{x}, \mathbf{u}, e) = q$, where it is possible that $q' = q$ (a self-loop transition). That is, the reset condition is such that $\mathbf{r}(q', q, e, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}) \notin Inv(q)$. In case (a), $g_i(\cdot)$ is such that $g_i(t, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}) = 0$ is part of the boundary of $Inv(q)$ including $(\mathbf{x}, \mathbf{u})$. In case (b), the transition can only occur as (i) a result of some $e \in \mathcal{E}$ (completing the proof); (ii) the violation of $Inv(q')$; or (iii) while $(\mathbf{x}, \mathbf{u}) \in guard(q', q)$. Since we have already considered the first two cases, we only need to check case (iii), including $q' = q$. This case can occur either through (A) a policy equivalent to a condition $g(t, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}) = 0$; or (B) after some random time. In the former case, we can define some E_i such that $g_i(t, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}) = g(t, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}) = 0$ and include $E_i \in \tilde{\mathcal{E}}$. In case (B), let $\tau = \inf\{t \geq \tau_{k-1} : (\mathbf{x}(t), \mathbf{u}(t)) \in guard(q', q)\}$ be the time that $(\mathbf{x}, \mathbf{u})$ enters $guard(q', q)$. We can associate a self-loop transition (q', q') at τ which is caused by some event E_i with guard function $g_i(\cdot)$ such that $g_i(\tau, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}) = 0$. Note that $(\mathbf{x}, \mathbf{u})$ satisfying this condition forms part of the boundary of $guard(q, q')$. We define a reset condition for this transition such that a timer with state y gets initialized with a random value $V_i(\boldsymbol{\theta})$ such that $\tau + V_i(\boldsymbol{\theta})$ is the time of transition (q', q). We can then include y in $\tilde{\mathbf{x}}$ and define the event at $\tau + V_i(\theta)$ as $E_j \in \tilde{\mathcal{E}}$ and associate with it a guard function $g_j(t, \tilde{\mathbf{x}}, \mathbf{u}, \boldsymbol{\theta}) = y_j$. Since such events E_i, E_j can always be defined, the proof is complete. ∎

In the above proof, we turn the reader's attention to how timers and guard functions remove the need for guard sets. Also, in the case of a chain of simultaneous transitions, they identify an event whose guard function determines when the transitions occur.

Example. To illustrate this framework based on events associated with guard functions, consider the example of a SHA as shown in Fig. 2(a). This models a simple flow system with a buffer whose content is $x(t, \theta) \geq 0$. The dynamics of the content are $\dot{x}(t, \theta) = 0$ when $q = 0$ (empty buffer) and $\dot{x}(t, \theta) = \alpha(t, \theta) - \beta$, otherwise. Here, $\beta \geq 0$ is a fixed outflow rate and $\{\alpha(t, \theta)\}$ is a piecewise differentiable random process whose behavior depends on θ via a continuous vector field $f_\alpha(t, \theta)$. We allow for discontinuous jumps in the value of $\alpha(t, \theta)$ at random points in time modeled through events that occur at time instants $V_1, V_1 + V_2, \ldots$ using a timer state variable $y(t)$ re-initialized to V_{n+1} after the nth time that $y(t) = 0$. The result of such an event at time t is a new value $\alpha(t^+) = A_{n+1}$ where this jump process is independent of θ. In mode $q = 0$, the invariant condition $\alpha(t, \theta) \leq \beta$ is required to ensure that the buffer remains empty.

The state of the new SHA, shown in Fig. 2(b), is denoted by $\mathbf{x} = (\alpha, x, y)$ where note that $\dot{y}(t) = -1$. The event set is $\mathcal{E} = \{E_1, E_2, E_3\}$ with $g_1(t, \tilde{\mathbf{x}}, \mathbf{u}, \boldsymbol{\theta}) = \alpha(t, \theta) - \beta$, $g_2(t, \tilde{\mathbf{x}}, \mathbf{u}, \boldsymbol{\theta}) = x(t, \theta)$, and $g_3(t, \tilde{\mathbf{x}}, \mathbf{u}, \boldsymbol{\theta}) = y(t, \theta)$. In addition, we define the reset condition $\mathbf{r}(0, 0, E_3) = \mathbf{r}(1, 1, , E_3) = (A_{n+1}, x, V_{n+1})$ whenever E_3 occurs for the nth time, treating $\{A_n\}$ and $\{V_n\}$, $n = 1, 2, \ldots$, as input processes.

When $q = 0$, two events are possible: (i) If E_1 occurs at $t = \tau_k$ we have $\alpha(\tau_k, \theta) - \beta = 0$ and a transition to $q = 1$ occurs since the condition $\alpha(t, \theta) - \beta < 0$ must have held at τ_k^-. (ii) If E_3 occurs at $t = \tau_k$ we have $y(\tau_k) = 0$ and a self-loop transition results. By the reset condition, $\alpha(\tau_k^+) = A_{n+1}$ for some A_{n+1}, assuming this was the nth occurrence of this event. If $A_{n+1} > \beta$, then immediately a transition to $q = 1$ occurs. Observe that even though the condition of this transition is $\alpha(\tau_k^+) > \beta$, the transition is still due to event E_3.

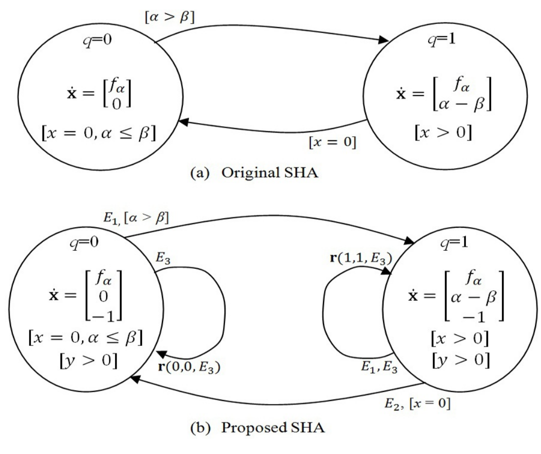

Fig. 2. A simple fluid buffer SHA: Contrasting two approaches.

When $q = 1$, all three events are possible, but E_1,E_3 cause self loops. If E_2 occurs at some $t = \tau_k$, then $x(\tau_k) = 0$. On the other hand, suppose that a transition to $q = 1$ occurred because of E_1 at $t = \tau_k$, i.e., $\alpha(\tau_k, \theta) - \beta = 0$. It is possible that $\alpha(t, \theta) - \beta = 0$ for $t \in [\tau_k, \tau_k + \epsilon]$, $\epsilon > 0$.

In this case, $\dot{x} = \alpha(t, \theta) - \beta = 0$ and $x(\tau_k^+) = 0$. This violates the invariant condition $[x > 0]$ at $q = 1$, causing an immediate return to $q = 0$. Similarly, if $\alpha(\tau_k, \theta) - \beta = 0$ but $\alpha(\tau_k^+, \theta) - \beta < 0$, the invariant condition at $q = 1$ is violated and there is an immediate return to $q = 0$. All these can be summarized in the transition functions below:

$$\phi(0, \mathbf{x}, \mathbf{u}, e) = \begin{cases} 1 & \text{if } e = E_1 \text{ or } \alpha(t, \theta) > \beta \\ 0 & \text{otherwise} \end{cases}$$

$$\phi(1, \mathbf{x}, \mathbf{u}, e) = \begin{cases} 0 & \text{if } e = E_2 \text{ or } x(t, \theta) = 0 \\ 1 & \text{otherwise} \end{cases}$$

where $\mathbf{u} = \alpha$.

It is important to note that the conditions $[\alpha > \beta]$, $[x = 0]$ have different meanings in Figs.2(a),(b). In the former, they define the guard set conditions. As confirmed by Fig.2(a), the guard set conditions (e.g. , $[\alpha > \beta]$ at $q = 0$) cannot differentiate between (i) a smooth transition from the invariant $[\alpha \leq \beta]$ to $[\alpha > \beta]$, hence, a transition to $q = 1$ and (ii) a jump in $\alpha(t)$ that causes $[\alpha(t^+) > \beta]$ to become true without satisfying $\alpha(t) = \beta$. Recall that the former depends on θ, whereas the latter does not. On the other hand, the set conditions on the transitions in Fig.2(b) have a different meaning: they identify a condition caused by an event occurring at the same time but in a previous transition. Thus, $[\alpha > \beta]$ is clearly associated with a jump in α in the previous transition and is independent of θ. If we are to control θ to affect the systems's performance, it is obviously crucial to identify transitions that depend on it as opposed to ones that do not.

Another byproduct of using the guard functions is that unlike the conventional IPA approach which normally categorizes the events into *exogenous, endogenous,* and *induced* classes and derives different equations to capture their dependence on parameter vector $\boldsymbol{\theta}$, guard functions enable us to treat all events uniformly. Moreover, inclusion of timers in the states eliminates the need for a spontaneous event and the ambiguous notions of "enabled" events and "waiting" in a guard set. To show that the framework above encompasses the event classification in [Cassandras et al., 2009] for a SFM, we give a simple definition of each event class in terms of guard functions:

• *Exogenous Events*: In [Cassandras et al., 2009], an event $E_i \in \mathcal{E}$ is defined as exogenous if it causes a transition at time τ_k independent of $\boldsymbol{\theta}$ and satisfies the gradient condition $\frac{d\tau_k}{d\boldsymbol{\theta}} = \mathbf{0}$. In our case, we define E_i occurring at $t = \tau_k$ as exogenous if the associated guard function $g_i(\tau_k, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}) = g_i(\tau_k, \mathbf{x}, \mathbf{u})$ is independent of $\boldsymbol{\theta}$ so that $\frac{dg_i(\tau_k, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta})}{d\boldsymbol{\theta}} = \mathbf{0}$. In our framework, an exogenous event is associated with a timer state variable whose guard function is independent of $\boldsymbol{\theta}$.

• *Endogenous Events*: In contrast to an exogenous event, an endogenous one is such that $\frac{dg_i(\tau_k, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta})}{d\boldsymbol{\theta}} \neq \mathbf{0}$. This includes cases where $g_i(\tau_k, \mathbf{x}(\tau_k, \boldsymbol{\theta}), \mathbf{u}(\tau_k, \boldsymbol{\theta}), \boldsymbol{\theta}) = y_i(\tau_k, \boldsymbol{\theta}) = 0$, for some timer state y_i.

• *Induced Events*: In [Cassandras et al., 2009], an event $E_i \in \mathcal{E}$ occurring at $t = \tau_k$ is called *induced* if it is caused by the occurrence of another event (the *triggering* event) at $\tau_m < \tau_k$ $(m < k)$. In our case, an event is induced if there exists a state variable x associated with $E_j \in \mathcal{E}$ for which the following conditions are met:

$$\tau_m = \max\{t < \tau_k : x(t, \boldsymbol{\theta}) \neq 0,\ \forall t \in (\tau_m, \tau_k)\}. \quad (1a)$$

$$\tau_k = \min\{t > \tau_m : x(t, \boldsymbol{\theta}) = 0\}. \quad (1b)$$

In this case, the active guard function at τ_k is

$$g_j(\tau_k, \mathbf{x}(\tau_k, \boldsymbol{\theta}), \mathbf{u}(\tau_k, \boldsymbol{\theta}), \boldsymbol{\theta}) = x(\tau_k, \boldsymbol{\theta}) = 0. \quad (2)$$

Generally, the initial value of state $x(\tau_m, \boldsymbol{\theta})$ is determined by a reset function associated with transition m. After the reset, the dynamics $\dot{x} = f(t, \boldsymbol{\theta})$ can be arbitrary until $x(t, \boldsymbol{\theta}) = 0$ is satisfied. In this sense, the timer events are simple cases of induced events with trivial dynamics.

As already mentioned, it is possible to have simultaneous transitions. This is a necessary condition to have chattering in the SHA sample path which is mostly undesirable. To ensure a bounded number of transitions in the interval $[0, T]$, let us introduce the following assumption:

Assumption 3. With probability 1, the number of simultaneous transitions is finite.

Since the number of transitions occurring at different times is finite over a finite interval $[0, T]$, Assumption 3 ensures that the total number of events observed on the sample path is finite. Depending on the system in question Assumption 3 translates into conditions on the states, parameters and inputs (see Assumption 5 for the case example in Section 5). We do not give the conditions under which Assumption 3 is valid in a general SHS setting. More on this can be found in [Simić et al., 2000] and standard references on control of hybrid systems.

2.1 The optimization problem

Observing the SHS just described over the interval $[0, T]$, we seek to solve the following optimization problem:

$$(\mathbf{P}) \quad \boldsymbol{\theta}^* = \operatorname{argmin}_{\boldsymbol{\theta}}\ J(T, \boldsymbol{\theta}) = \mathrm{E}_\omega\big[L(T, \boldsymbol{\theta}, \omega)\big],$$

subject to

$$\begin{aligned} &\dot{\mathbf{x}}(t, \boldsymbol{\theta}, \omega) = \mathbf{f}(q(t), t, \mathbf{x}(t, \boldsymbol{\theta}, \omega), \mathbf{u}(t, \boldsymbol{\theta}, \omega), \boldsymbol{\theta}), \\ &\mathbf{x} \in Inv(q), \mathbf{u} \in \mathcal{U}(\boldsymbol{\theta}) \\ &\mathbf{x}(\tau_k^+, \boldsymbol{\theta}, \omega) = \mathbf{r}(q_{k-1}, q_k, \mathbf{x}(\tau_k, \boldsymbol{\theta}, \omega), \mathbf{u}(\tau_k, \boldsymbol{\theta}, \omega), \boldsymbol{\theta}) \\ &q(t) \in \{1, \ldots, N_q\},\ k = 1, \ldots, K(\omega) \\ &\mathbf{x}(0, \boldsymbol{\theta}, \omega) = \mathbf{x}_0, \end{aligned}$$

where $q(t) = q_k$ when $t \in [\tau_k, \tau_{k+1})$ and $L(T, \boldsymbol{\theta}, \omega)$ is a sample function generally defined as

$$L(T, \boldsymbol{\theta}, \omega) = \int_0^T \ell(q(t), t, \mathbf{x}(t, \boldsymbol{\theta}, \omega), \mathbf{u}(t, \boldsymbol{\theta}, \omega), \boldsymbol{\theta}) dt \quad (3)$$

for some given function $\ell(\cdot)$. Notice that although it is possible to treat time t as a continuous state variable, we make the dependence of various function on t explicit and do not include it in $\mathbf{x}$ as it makes our analysis easier to follow.

We solve problem $(\mathbf{P})$ using IPA. The objective of IPA is to specify how the changes in $\boldsymbol{\theta}$ affect $\mathbf{x}(t, \boldsymbol{\theta}, \omega)$ and ultimately, to calculate $\frac{dL(T, \boldsymbol{\theta}, \omega)}{d\boldsymbol{\theta}}$. This is done by finding the gradients of state $\mathbf{x}$ and event times $\tau_k(\boldsymbol{\theta})$,,$k = 1, \ldots, K$, with respect to $\boldsymbol{\theta}$. It has been shown that under mild smoothness conditions the result is an unbiased estimate of the objective function gradient $\frac{dJ(T, \boldsymbol{\theta})}{d\boldsymbol{\theta}}$ [Cassandras et al., 2009]. Thus, coupling the sensitivity estimates with a gradient-based optimization algorithm can optimize the system performance.

3. UNIFIED IPA APPROACH

3.1 Matrix Notation

Let $v(t,\boldsymbol{\theta})$ be a scalar function which is differentiable with respect to $\boldsymbol{\theta}$. We define the gradient vector with respect to $\boldsymbol{\theta}$ as $v'(t,\boldsymbol{\theta}) = \frac{\partial v(t,\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \big(\frac{\partial v(t,\boldsymbol{\theta})}{\partial \theta_1}, \dots, \frac{\partial v(t,\boldsymbol{\theta})}{\partial \theta_{N_\theta}}\big)$. Moreover, we denote the full and partial Jacobian of a vector $\mathbf{v}(t,\boldsymbol{\theta}) \in \mathbb{R}^M$ with respect to $\boldsymbol{\theta}$ by

$$\frac{d\mathbf{v}(t,\boldsymbol{\theta})}{d\boldsymbol{\theta}} = \left[\frac{dv_i(t,\boldsymbol{\theta})}{d\theta_j}\right] \in \mathbb{R}^{M\times N_\theta}, \tag{4}$$

$$\mathbf{v}'(t,\boldsymbol{\theta}) = \left[\frac{\partial v_i(t,\boldsymbol{\theta})}{\partial \theta_j}\right] \in \mathbb{R}^{M\times N_\theta} \tag{5}$$

where $v_i(t,\boldsymbol{\theta})$ $(i < M)$ is the i-th entry of $\mathbf{v}(t,\boldsymbol{\theta})$. With a slight abuse of notation we use $\frac{d\mathbf{v}(t,\boldsymbol{\theta})}{d\mathbf{x}} = \left[\frac{dv_i(t,\boldsymbol{\theta})}{dx_j}\right]$, $\frac{\partial \mathbf{v}(t,\boldsymbol{\theta})}{\partial \mathbf{x}} = \left[\frac{\partial v_i(t,\boldsymbol{\theta})}{\partial x_j}\right] \in \mathbb{R}^{M\times N_x}$ and $\frac{d\mathbf{v}(t,\boldsymbol{\theta})}{d\mathbf{u}} = \left[\frac{dv_i(t,\boldsymbol{\theta})}{du_k}\right]$, $\frac{\partial \mathbf{v}(t,\boldsymbol{\theta})}{\partial \mathbf{u}} = \left[\frac{\partial v_i(t,\boldsymbol{\theta})}{\partial u_k}\right] \in \mathbb{R}^{M\times N_u}$ as the full and partial Jacobians of $\mathbf{v}(t,\boldsymbol{\theta})$ with respect to $\mathbf{x}$ and $\mathbf{u}$.

For the event times $\tau_k(\boldsymbol{\theta})$, $k = 1,\dots,K$, the gradient with respect to $\boldsymbol{\theta}$ is defined as

$$\tau_k' = (\tau_{k,1}', \dots, \tau_{k,N_\theta}')$$

where $\tau_{k,j}' = \frac{\partial \tau_k}{\partial \theta_j}$. We let $\tau_{0,j}' = \tau_{K+1,j}' = 0$ for all j since the start and end of the sample path are fixed values. Finally, we define $\boldsymbol{\tau}'$ as a $N_e \times N_\theta$ matrix such that its ith row is associated with event E_i and its jth column is associated with the variable with respect to which the differentiation is done.

In what follows, we derive a unified set of equations which give the event-time and state derivatives with respect to $\boldsymbol{\theta}$ and are in concord with the results of IPA presented in [Cassandras et al., 2009]. All calculations can be done in two generic steps, regardless of the type of event observed, i.e., we do not need to differentiate between exogenous, endogenous, and induced events.

4. INFINITESIMAL PERTURBATION ANALYSIS

Below, we drop T from $L(T,\boldsymbol{\theta},\omega)$ and ω from the arguments of other functions to simplify the notation. However, we still write $L(\boldsymbol{\theta},\omega)$ to stress that we carry out the analysis on the sample path of system G denoted by ω. We write (3) as

$$L(\boldsymbol{\theta},\omega) = \sum_{k=0}^{K} \int_{\tau_k}^{\tau_{k+1}} \ell(q_k, t, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}) dt,$$

Recalling $\tau_0' = \tau_{K+1}' = \mathbf{0}$, we calculate the gradient of the sample cost with respect to $\boldsymbol{\theta}$ as

$$\begin{aligned}\frac{dL(\boldsymbol{\theta},\omega)}{d\boldsymbol{\theta}} &= \sum_{k=1}^{K} [\ell(q_{k-1}, \tau_k^-, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}) - \ell(q_k, \tau_k^+, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta})]\tau_k' \\ &+ \sum_{k=0}^{K} \int_{\tau_k}^{\tau_{k+1}} \frac{d\ell(q_k, t, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta})}{d\boldsymbol{\theta}} dt \end{aligned}\tag{6}$$

where

$$\begin{aligned}\frac{d\ell(q_k, t, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta})}{d\boldsymbol{\theta}} &= \frac{\partial \ell}{\partial \mathbf{x}} \frac{d\mathbf{x}(t,\boldsymbol{\theta})}{d\boldsymbol{\theta}} + \frac{\partial \ell}{\partial \mathbf{u}} \mathbf{u}'(t,\boldsymbol{\theta}) \\ &+ \ell'(q_k, t, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta})\end{aligned}$$

Thus, as mentioned before, in order to determine the sample cost gradient with respect to $\boldsymbol{\theta}$, one needs to find the event time and state derivatives with respect to it.

4.1 Event-time Derivatives

By Theorem 1, for any $k = 1,\dots,K$, transition k is directly or indirectly associated with an event $e \in \tilde{\mathcal{E}}$ with a guard function $g(\cdot)$ such that $g(\tau_k^-, \tilde{\mathbf{x}}, \mathbf{u}, \boldsymbol{\theta}) = 0$. Let us define the guard vector

$$\mathbf{g}(t,\mathbf{x},\mathbf{u},\boldsymbol{\theta}) = \big(g_1(t,\mathbf{x},\mathbf{u},\boldsymbol{\theta}), \dots, g_{N_e}(t,\mathbf{x},\mathbf{u},\boldsymbol{\theta})\big). \tag{7}$$

and a unit *firing vector* $\mathbf{e}_i$, $i = 1,\dots,N_e$ as

$$\mathbf{e}_i = \big(0, \dots, 1, \dots, 0\big) \in \mathbb{R}^{N_e}$$

where only the i-th element is 1 and the rest are 0.

Let $\mathbf{G}(t,\mathbf{x},\mathbf{u},\boldsymbol{\theta})$ be a diagonal matrix function where $\mathbf{G}_{i,i} = g_i$, $i = 1,\dots,N_e$ and denote its time derivative at $t = \tau$ by $\dot{\mathbf{G}}(\tau,\mathbf{x},\mathbf{u},\boldsymbol{\theta})$. We can obtain $\boldsymbol{\tau}'$ by differentiating $\mathbf{g}(\tau,\mathbf{x},\mathbf{u},\boldsymbol{\theta}) = \mathbf{0}$ with respect to $\boldsymbol{\theta}$. This gives

$$\boldsymbol{\tau}' = -\dot{\mathbf{G}}(\tau^-,\mathbf{x},\mathbf{u},\boldsymbol{\theta})^{-1} \frac{d\mathbf{g}(\tau^-,\mathbf{x},\mathbf{u},\boldsymbol{\theta})}{d\boldsymbol{\theta}} \tag{8}$$

where

$$\begin{aligned}\frac{d\mathbf{g}(\tau^-,\mathbf{x},\mathbf{u},\boldsymbol{\theta})}{d\boldsymbol{\theta}} &= \frac{\partial \mathbf{g}}{\partial \mathbf{x}} \frac{d\mathbf{x}(\tau^-,\boldsymbol{\theta})}{d\boldsymbol{\theta}} \\ &+ \frac{\partial \mathbf{g}}{\partial \mathbf{u}} \mathbf{u}'(\tau^-,\boldsymbol{\theta}) + \mathbf{g}'(\tau^-,\mathbf{x},\mathbf{u},\boldsymbol{\theta}).\end{aligned}\tag{9}$$

It is easily verified that the simple equation (8) is in line with what has been reported in prior work on IPA for SHS, e.g., in [Cassandras et al., 2009]. The following assumption is introduced so that τ_k' exists:

Assumption 4. With probability 1, if τ_k is the occurrence time of E_i, we have $\dot{g}_i(\tau_k,\mathbf{x},\mathbf{u},\boldsymbol{\theta}) \neq 0$.

Note that the case of a contact point where $\dot{g}_i(\tau_k,\boldsymbol{\theta})$ does not exist has already been excluded by Assumption 2, hence, $\dot{g}_i(\tau_k,\mathbf{x},\mathbf{u},\boldsymbol{\theta})$ is always well-defined. Also, observe that only one row in (8) is evaluated at each transition. In fact, $\boldsymbol{\tau}'$ is a generic matrix function such that if transition k is associated with event E_i, only its ith row is evaluated.

4.2 State Derivatives

Here, we determine how the state derivatives evolve both at transition times and in between them, i.e., within a mode $q \in Q$.

Derivative update at transition times: At each transition we consider two cases:

(a) *No Reset*: In this case, we have $x_j(\tau_k^-,\boldsymbol{\theta}) = x_j(\tau_k^+,\boldsymbol{\theta})$ for all $j \in \{1,\dots,N_x\}$. Assume that for every $q \in Q$, $\dot{x}_j(t,\boldsymbol{\theta}) = f_j(q,t,\mathbf{x},\mathbf{u},\boldsymbol{\theta})$. Then, we use the following equation, derived in [Cassandras et al., 2009], to update the state derivatives:

$$\begin{aligned}x_j'(\tau_k^+,\boldsymbol{\theta}) &= x_j'(\tau_k^-,\boldsymbol{\theta}) \\ &+ [f_j(q_{k-1},\tau_k^-,\mathbf{x},\mathbf{u},\boldsymbol{\theta}) - f_j(q_k,\tau_k^+,\mathbf{x},\mathbf{u},\boldsymbol{\theta})]\tau_k'.\end{aligned}\tag{10}$$

(b) *Reset*: In this case, there exists $j \in \{1,\dots,N_x\}$ such that $x_j(\tau_k^-,\boldsymbol{\theta}) \neq x_j(\tau_k^+,\boldsymbol{\theta})$. Let $e \in \mathcal{E}$ be the direct cause of transition (q_{k-1}, q_k) at τ_k (i.e., e appears on the arc connecting q_{k-1} and q_k in the automaton). We then define

a reset condition $x_j(\tau_k^+, \boldsymbol{\theta}) = r_j(q_{k-1}, q_k, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}, e)$ where $r_j(\cdot)$ is the reset function of x_j. Thus, we get

$$x'_j(\tau_k^+, \boldsymbol{\theta}) = \frac{dr'_j(q_{k-1}, q_k, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}, e)}{d\boldsymbol{\theta}} \tag{11}$$

where

$$\begin{aligned}\frac{dr'_j(q_{k-1}, q_k, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}, e)}{d\boldsymbol{\theta}} =& \frac{\partial r_j}{\partial \mathbf{x}}\mathbf{x}'(\tau_k, \boldsymbol{\theta}) + \frac{\partial r_j}{\partial \mathbf{u}}\mathbf{u}'(\tau_k, \boldsymbol{\theta}) \\ &+ r'_j(q_{k-1}, q_k, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}, e).\end{aligned} \tag{12}$$

For other transitions which are indirectly caused by an event, we simply define $\mathbf{r}(q_{k-1}, q_k, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}) = \mathbf{x}(\tau_k, \boldsymbol{\theta})$ as no reset is possible on them.

To put everything in matrix form, let us first define, for every $i = 1, \ldots, N_e$, the index set

$\Phi_i = \{(m, n) \in Q \times Q : \exists \mathbf{x}, \mathbf{u} \in \mathcal{X}, \mathcal{U} \text{ s.t.} \phi(m, \mathbf{x}, \mathbf{u}, E_i) = n\}$

containing all transitions directly associated with E_i. Also, for each i, let us define the reset mapping

$$\mathbf{r}_i(m, n) = \begin{cases} \mathbf{r}(m, n, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}, E_i) & \text{if } (m, n) \in \Phi_i \\ \mathbf{x} & \text{otherwise} \end{cases}$$

and the diagonal matrix $\mathbf{C}(m, n) \in \mathbb{R}^{N_x \times N_x}$ with its jth diagonal entry $c_{jj}(m, n) = 1$ if x_j is not reset by transition (m, n) and 0, otherwise. We also define $\bar{\mathbf{C}}(m, n) = \mathbf{I}_{N_x \times N_x} - \mathbf{C}(m, n)$ where $\mathbf{I}_{N_x \times N_x}$ is the identity matrix with the specified dimensions. Moreover, let us define the reset map matrix as

$$\mathbf{R}(\tau_k) = \begin{bmatrix} \mathbf{r}_1(q_{k-1}, q_k) \\ \vdots \\ \mathbf{r}_{N_e}(q_{k-1}, q_k) \end{bmatrix} \in \mathbb{R}^{N_e \times N_x} \tag{13}$$

where $x_j(\tau_k^+) = r_{i,j}(q_{k-1}, q_k)$ when the kth transition is due to E_i and $x_j(\tau_k^+) = x_j(\tau_k^-)$, otherwise. Thus, if $\mathbf{x}$ remains continuous at its occurrence time we get $\mathbf{r}_i(q_{k-1}, q_k) = \mathbf{x}(\tau_k, \boldsymbol{\theta})$. Finally, we define the shorthand notation $\Delta\mathbf{f}(q_{k-1}, q_k, \boldsymbol{\theta}) \equiv \mathbf{f}(q_{k-1}, \tau_k^-, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}) - \mathbf{f}(q_k, \tau_k^+, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta})$ for the jump in the dynamics at the kth transition. Using these definitions we can combine part (a) and (b) above and write

$$\begin{aligned}\mathbf{x}'(\tau_k^+) = \mathbf{C}(q_{k-1}, q_k)\big[\mathbf{x}'(\tau_k^-) + \Delta\mathbf{f}(q_{k-1}, q_k, \boldsymbol{\theta})^\mathsf{T}\tau'_k\big] \\ + \big(\mathbf{e}_i\mathbf{R}(\tau_k)\bar{\mathbf{C}}(q_{k-1}, q_k)\big)'.\end{aligned} \tag{14}$$

Derivative update between transitions: Assuming the mode is q_k, we only need to perform the following operation on interval $[\tau_k, \tau_{k+1})$:

$$\mathbf{x}'(t, \boldsymbol{\theta}) = \mathbf{x}'(\tau_k^+, \boldsymbol{\theta}) + \int_{\tau_k}^{t} \frac{d\mathbf{f}(q_k, \tau, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta})}{d\boldsymbol{\theta}} d\tau \tag{15}$$

where $\frac{d\mathbf{f}(q_k, \tau, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta})}{d\boldsymbol{\theta}}$ is a $N_x \times N_\theta$ Jacobian matrix of the state dynamics defined on $[\tau_k, \tau_{k+1})$ as

$$\begin{aligned}\frac{d\mathbf{f}(q_k, \tau, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta})}{d\boldsymbol{\theta}} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}\frac{d\mathbf{x}(\tau, \boldsymbol{\theta})}{d\boldsymbol{\theta}} + \frac{\partial \mathbf{f}}{\partial \mathbf{u}}\mathbf{u}'(\tau, \boldsymbol{\theta}) \\ + \mathbf{f}'(q_k, \tau, \mathbf{x}, \mathbf{u}, \boldsymbol{\theta}).\end{aligned} \tag{16}$$

To sum up, the basis for IPA on a system modeled as G is the pre-calculation of the quantities $\frac{\partial \ell}{\partial \mathbf{x}}, \frac{\partial \ell}{\partial \mathbf{u}}, \frac{\partial \mathbf{f}}{\partial \mathbf{x}}, \frac{\partial \mathbf{f}}{\partial \mathbf{u}}$ for all $q \in Q$, $\frac{\partial \mathbf{r}_i}{\partial \mathbf{x}}, \frac{\partial \mathbf{r}_i}{\partial \mathbf{u}}$ and $\frac{\partial \mathbf{g}}{\partial \boldsymbol{\theta}}$, which are then used in (8), (14) and (15) to update IPA derivatives. Finally, the results are applied to (6). We will next apply this method to a specific problem of interest [Cassandras and Lafortune, 2006] in the following section.

5. IPA FOR A SINGLE-NODE SFM

In what follows, we apply the method described above to a simple single-class single-node system shown in Fig.3. We use a simplified notation here for space limitations.

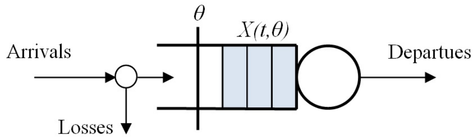

Fig. 3. The DES model of the single node system.

However, the dependence of functions on their arguments should be clear from the analysis in the previous section.

The system consists of a queue whose content level $X(t, \theta)$ is subject to stochastic arrival and service time processes. The queue capacity is limited to a quantity θ treated as the control parameter. Every arrival seeing a full queue is lost and incurs a penalty. Considering this system over a finite interval $[0, T]$, we want to find the best θ to trade off between the average workload and average loss defined as

$$\mathrm{E}_\omega[Q_{DES}(T, \theta, \omega)] = \frac{1}{T}\mathrm{E}_\omega\left[\int_0^T X(t, \theta, \omega)dt\right], \tag{17}$$

$$\mathrm{E}[L_{DES}(T, \theta, \omega)] = \frac{1}{T}\mathrm{E}_\omega N_{loss}(T, \theta, \omega) \tag{18}$$

where $N_{loss}(\cdot)$ is the number of losses observed in the interval $[0, T]$. Even for a simple system like this, the analysis can become prohibitive when the stochastic processes considered are arbitrary. Use of SFMs has proven to be very helpful in the analysis and optimization of queuing systems such as this one [Cassandras and Lafortune, 2006],[Cassandras et al., 2002] where applying IPA has resulted in very simple derivative estimates of the loss and workload objectives with respect to θ. In the SFM, the arrivals and departures are abstracted into non-negative stochastic inflow rate $\{\alpha(t)\}$ and maximal service rate $\{\beta(t)\}$ processes which are independent of θ. These rates continuously evolve according to the differential equations $\dot{\alpha} = f_\alpha(t)$ and $\dot{\beta} = f_\beta(t)$ where f_α and f_β are arbitrary continuous functions. It is important to observe that the precise nature of these functions turns out to be irrelevant as far as the resulting IPA estimator is concerned: the IPA estimators are independent of f_α and f_β. This is an important robustness property of IPA estimators which holds under certain conditions [Yao and Cassandras, 2011]. We allow for discrete jumps in both processes $\{\alpha(t)\}$ and $\{\beta(t)\}$ and use timer states $y_\alpha, y_\beta \geq 0$ to capture them. The fluid discharge rate $d(t, \theta)$ is defined as $d(t, \theta) = \beta(t)$ when $x(t, \theta) > 0$ and $d(t, \theta) = \alpha(t)$ otherwise. For this example, Assumption 3 manifests itself as follows:

Assumption 5. With probability 1, condition $\alpha(t) = \beta(t)$ cannot be valid on a non-empty interval containing t.

The buffer content process evolves according to the differential equation $\dot{x}(t, \theta) = \alpha(t) - d(t, \theta)$ so we can write

$$\dot{x}(t, \theta) = f_x(t, \theta) = \begin{cases} 0 & \text{if } x(t, \theta) = 0 \text{ or } \theta \\ \alpha(t) - \beta(t) & \text{otherwise} \end{cases} \tag{19}$$

When $x(t, \theta)$ reaches the buffer capacity level θ, a portion of the incoming flow is rejected with rate $\alpha(t) - \beta(t) > 0$.

Obviously, when $x(t,\theta) < \theta$ no loss occurs. Hence, we define, for every $t \in [0,T]$, the *loss rate* as

$$\ell(t,\theta) = \begin{cases} 0 & \text{if } x(t,\theta) < \theta \\ \alpha(t) - \beta(t) & \text{otherwise} \end{cases} \tag{20}$$

The SHA of this system is shown in Fig. 4. We define the

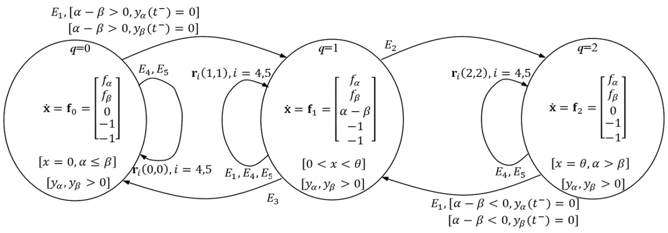

Fig. 4. Stochastic Flow Model (SFM) for the single-class SFM.

system state vector $\mathbf{x} = (\alpha, \beta, x, y_\alpha, y_\beta)$ and the input $\mathbf{u} = (\{A_j\}, \{V_j^\alpha\}, \{B_k\}, \{V_k^\beta\})$ whose elements are sequences of random variables from the jump distributions associated with states α, y_α, β, and y_β, respectively. Although it follows from the definitions that the states α, β, y_α, y_β are independent of θ yielding $\alpha'(t) = \beta'(t) = y'_\alpha(t) = y'_\beta(t) = 0$ for all $t \in [0,T]$, we include them at the start of the IPA estimation procedure to fully illustrate the matrix setting we have developed and make use of it later. Regarding the dynamics of the states, we have

$$\mathbf{f}_q = \begin{cases} (f_\alpha, f_\beta, 0, -1, -1) & \text{if } q = 0 \\ (f_\alpha, f_\beta, \alpha - \beta, -1, -1) & \text{if } q = 1 \\ (f_\alpha, f_\beta, 0, -1, -1) & \text{otherwise} \end{cases} \tag{21}$$

The workload and loss sample functions are

$$Q(T,\theta,\omega) = \frac{1}{T}\int_0^T x(t,\theta,\omega)dt,$$

$$L(T,\theta,\omega) = \frac{1}{T}\int_0^T \ell(t,\theta,\omega)dt.$$

Using the fact that $x(t,\theta)$ and $\ell(t)$ can only contribute to their associated objectives when, respectively, $x > 0$ ($q = 1,2$) and $x = \theta$ ($q = 2$), we can write:

$$Q(T,\theta,\omega) = \frac{1}{T}\sum_{n=1}^{N}\int_{\xi_n(\theta)}^{\eta_n(\theta)} x(t,\theta,\omega)dt, \tag{22}$$

$$L(T,\theta,\omega) = \frac{1}{T}\sum_{n=1}^{N}\sum_{m=1}^{M_n}\int_{\nu_{n,m}(\theta)}^{\sigma_{n,m}(\theta)} \ell(t,\theta,\omega)dt \tag{23}$$

where N is the number of supremal intervals $[\xi_n, \eta_n)$, $n = 1,\dots,N$ over which $x(t,\theta,\omega) > 0$ (i.e., $q = 1,2$) and M_n is the number of supremal intervals $[\nu_{n,m}, \sigma_{n,m})$, $m = 1,\dots,M_n$ such that $x(t,\theta,\omega) = \theta$ (i.e., $q = 2$). We refer to the intervals of the first kind as *Non-Empty Periods* (NEPs) and the second kind as *Full Periods* (FPs). We also drop the sample path index ω to simplify the notation.

Differentiating (22) with respect to θ and noting $x(\xi_n(\theta)^+) = x(\eta_n(\theta)^-) = 0$ for any n gives

$$Q'(T,\theta) = \frac{1}{T}\sum_{n=1}^{N}\left[x(\eta_n(\theta)^-)\eta'_n(\theta) - x(\xi_n(\theta)^+)\xi'_n(\theta) + \int_{\xi_n(\theta)}^{\eta_n(\theta)} x'(t,\theta)dt\right] = \frac{1}{T}\sum_{n=1}^{N}\int_{\xi_n(\theta)}^{\eta_n(\theta)} x'(t,\theta)dt, \tag{24}$$

Next, differentiating (23) with respect to θ and noticing that by (20), $\ell'(t,\theta) = 0$ for all $t \in [0,T)$ (eliminating the integral part), reveals

$$L'(T,\theta) = \sum_{n=1}^{N}\sum_{m=1}^{M_n}[\sigma'_{n,m}\ell(\sigma^-_{n,m},\theta) - \nu'_{n,m}\ell(\nu^+_{n,m},\theta)]. \tag{25}$$

It is now clear that to evaluate (24) and (25) we only need to obtain $x'(t,\theta)$ for all $t \in [\xi_n, \eta_n)$, $n = 1,\dots,N$ and event time derivatives $\nu'_{n,m}, \sigma'_{n,m}$ for every $m = 1,\dots,M_n$ where $n = 1,\dots,N$. However, as mentioned before, we try to keep everything in the general matrix framework so as to verify its effectiveness.

According to Fig. 4, the event set of the SHS is given as

$$\mathcal{E} = \{E_i, i = 1,\dots,5\}.$$

with guard functions defined as follows: E_1 occurs when $g_1 = \alpha - \beta = 0$. E_2 is the event of reaching the buffer threshold θ, so that $g_2 = x - \theta$. E_3 is the event ending a non-empty period, hence $g_3 = x$. Finally, E_4 and E_4 are associated with the timer run-offs captured by $g_4 = y_\alpha$ and $g_5 = y_\beta$, respectively. To summarize, the guard vector for the system is

$$\mathbf{g}(t,\theta) = \big(\alpha(t) - \beta(t), x(t,\theta) - \theta, x(t,\theta), y_\alpha(t), y_\beta(t)\big). \tag{26}$$

The reset maps are defined as follows:

$$\mathbf{r}_i(m,n) = \begin{cases} (A_j, \beta, x, V_j^\alpha, y_\beta) & \text{if } i = 4, m = n \in \{0,1,2\} \\ (\alpha, B_k, x, y_\alpha, V_k^\beta) & \text{if } i = 5, m = n \in \{0,1,2\} \\ \mathbf{x} & \text{otherwise} \end{cases}$$

where A_j and B_k are, respectively, the jth and kth elements of random sequences $\{A_j\}$ and $\{B_k\}$. Using these results in (13) we get

$$\mathbf{R}(\tau_k) = \begin{bmatrix} \mathbf{x}(\tau_k) \\ \mathbf{x}(\tau_k) \\ \mathbf{x}(\tau_k) \\ \mathbf{r}_4(q_{k-1}, q_k) \\ \mathbf{r}_5(q_{k-1}, q_k) \end{bmatrix}. \tag{27}$$

By (14), only the reset conditions associated with discontinuous states need to be differentiated with respect to θ. Since x, the only state variable which depends of θ, is continuous, we conclude that the last term in (14) is always $\mathbf{0}$ and need not be evaluated.

The IPA starts by evaluating (8). Note that by (21) $\frac{d\mathbf{f}_q(t)}{d\theta} = \mathbf{0}$ for all $q \in \{0,1,2\}$. Thus, we only need to evaluate (9). Also, by definition, $\frac{\partial \mathbf{u}}{\partial \theta} = \mathbf{0}$, so $\frac{\partial \mathbf{g}}{\partial \mathbf{u}}$ need not be evaluated. Then, from (26), we are left with

$$\frac{\partial \mathbf{g}}{\partial \mathbf{x}} = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \ \mathbf{g}'(t,\theta) = \begin{bmatrix} 0 \\ -1 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

It follows from (9) that

$$\frac{d\mathbf{g}(\tau^-,\theta)}{d\theta} = \begin{bmatrix} \alpha'(\tau^-) - \beta'(\tau^-) \\ x'(\tau^-) - 1 \\ x'(\tau^-) \\ y'_\alpha(\tau^-) \\ y'_\beta(\tau^-) \end{bmatrix} = \begin{bmatrix} 0 \\ x'(\tau^-) - 1 \\ x'(\tau^-) \\ 0 \\ 0 \end{bmatrix}$$

where we used the fact that α, β, y_α and y_β are independent of θ. Moreover, by (26), we have the time derivative of the guard functions just before the kth transition as

$\dot{\mathbf{G}}(\tau^-,\theta) = \mathrm{diag}\big(f_\alpha(\tau^-) - f_\beta(\tau^-), \dot{x}(\tau^-), \dot{x}(\tau^-), -1, -1\big)$ and since E_2 and E_3 are only feasible when $x > 0$ ($q = 1,2$), by (19) we get $\dot{x}(\tau^-) = \alpha(\tau^-) - \beta(\tau^-) \neq 0$ in the above expression. Combining the results in (8) gives

$$\boldsymbol{\tau}' = -\left(0, \frac{x'(\tau^-) - 1}{\alpha(\tau^-) - \beta(\tau^-)}, \frac{x'(\tau^-)}{\alpha(\tau^-) - \beta(\tau^-)}, 0, 0\right)^{\mathsf{T}}. \quad (28)$$

Next, we determine the state derivatives with respect to θ. Since x is the only state dependent on θ, we only apply (14) and (15) to $x(t,\theta)$. We also need not evaluate $\mathbf{C}(q_{k-1}, q_k)$, $k = 1,\dots,K$ as $x(t,\theta)$ is continuous throughout $[0,T]$. Moreover, we need to determine $\Delta f_x(q_{k-1}, q_k)$ for any feasible transition (q_{k-1}, q_k). If we define $\lambda(\tau) = \alpha(\tau) - \beta(\tau)$ and $\Delta\lambda(\tau) = \lambda(\tau^-) - \lambda(\tau^+)$ we get

$$\Delta f_x(q_{k-1}, q_k) = \begin{cases} -\lambda(\tau_k^+) & \text{if } (q_{k-1}, q_k) \in \{(0,1),(2,1)\} \\ \lambda(\tau_k^-) & \text{if } (q_{k-1}, q_k) \in \{(1,0),(1,2)\} \\ \Delta\lambda(\tau_k) & \text{if } (q_{k-1}, q_k) = (1,1) \\ 0 & \text{otherwise} \end{cases}$$

Invoking (14) for $x(t,\theta)$ gives

$$x'(\tau_k^+,\theta) = x'(\tau_k^-,\theta) + \Delta f_x(q_{k-1}, q_k, \theta)\tau_k'. \quad (29)$$

By (28), we only need to take care of the transitions caused by E_2 and E_3 as in other cases $\Delta f_x(q_{k-1}, q_k, \theta)\tau_k' = 0$. Since neither E_2 nor E_3 appear in the transitions with resets, they cannot create a chain of simultaneous transitions, thereby leaving us with transition $(1,2)$ for E_2 and $(1,0)$ for E_3. In the first case, we get $\Delta f_x(1,2,\theta)\tau_k' = 1 - x'(\tau_k^-,\theta)$ and in the latter case we get $\Delta f_x(1,0,\theta)\tau_k' = -x'(\tau_k^-,\theta)$. Inserting these results in (29) yields

$$x'(\tau_k^+,\theta) = \begin{cases} 1 & \text{if } (q_{k-1}, q_k) = (1,2) \\ 0 & \text{if } (q_{k-1}, q_k) = (1,0) \\ x'(\tau_k^-,\theta) & \text{otherwise} \end{cases} \quad (30)$$

There is no need to consider (15) in this case, since $\frac{d\mathbf{f}_q}{d\theta} = 0$ for all q. Therefore, we are in the position to fully evaluate the sample derivative estimates (24) and (25).

By (30), after the buffer becomes empty (transition (1,0) through event E_3), $x'(t,\theta)$ becomes and stays at 0 until a transition $(1,2)$ occurs through E_2. If this happens, $x'(t,\theta)$ resets to 1 in (30) and remains constant until the buffer becomes empty again. Therefore, we need only consider those nonempty periods $[\xi_n, \eta_n)$ in which a transition $(1,2)$ occurs. If this happens, we calculate the length of the interval between the first such transition until the next time the buffer becomes empty. $Q'(T,\theta)$ in (24) is the sum of lengths of these intervals, i.e.,

$$Q'(T,\theta) = \frac{1}{T}\sum_{n=1}^{N} \mathbf{1}_{FP}(n)(\eta_n - \nu_{n,1}) \quad (31)$$

where $\mathbf{1}_{FP}(n) = 1$ if there exists a transition $(1,2)$ in the non-empty period $[\xi_n, \eta_n)$ and 0 otherwise.

Next, to evaluate (25), notice that at $t = \sigma_{n,m}$ (end of stay at $q = 2$ in Fig. 4) a transition to $q = 1$ can occur in two ways: (a) Through E_4 or E_5 (transition (2,2)) and violating the invariant condition $[\alpha > \beta]$ which immediately fires transition $(2,1)$; (b) Directly, by E_1 (transition (2,1)). These three possibilities are associated with zeros in (28), so we have $\sigma_{n,m}' = 0$. Regarding the term $-\nu_{n,m}'\ell(\nu_{n,m}^+,\theta)$ in (25), the event at $\nu_{n,m}$ is E_2. By (28), we have $\nu_{n,m}' = -\frac{x'(\nu_{n,m}^-,\theta)-1}{\alpha(\nu_{n,m}^-)-\beta(\nu_{n,m}^-)}$. Since by (20), $\ell(\nu_{n,m}^+,\theta) = \alpha(\nu_{n,m}^+) - \beta(\nu_{n,m}^+)$ and by Assumption 2, $\alpha(\nu_{n,m}^+) - \beta(\nu_{n,m}^+) = \alpha(\nu_{n,m}^-) - \beta(\nu_{n,m}^-)$, we find that $-\nu_{n,m}'\ell(\nu_{n,m}^+,\theta) = x'(\nu_{n,m}^-,\theta) - 1$. We have already shown in (30) that in a non-empty period $[\xi_n, \eta_n)$, $x'(t,\theta) = 0$ for all $t \in [\xi_n, \nu_{n,1})$ and $x'(t,\theta) = 1$, $t \in [\nu_{n,1}, \eta_n)$. Hence, $x'(\nu_{n,m}^-,\theta) = 0$ when $m = 1$ and $x'(\nu_{n,m}^-,\theta) = 1$, otherwise. Combining all results into (25), we find that

$$L'(T,\theta) = \sum_{n=1}^{N}\sum_{m=1}^{M_n} -\nu_{n,m}'\ell(\nu_{n,m}^+,\theta) = -N_F$$

where N_F is the number of non-empty intervals with at least one full period. These results recover those in [Cassandras and Lafortune, 2006, pp. 700–703] and [Cassandras et al., 2002]. Note that $Q'(T,\theta)$ and $L'(T,\theta)$ are independent of f_α and f_β, i.e., these sensitivity estimates are independent of the random arrival and service processes, a fundamental robustness property of IPA.

6. CONCLUSIONS

We have introduced a general framework suitable for analysis and on-line optimization of Stochastic Hybrid Systems (SHS) which facilitates the use of Infinitesimal Perturbation Analysis (IPA). In doing so, we modified the previous structure of a Stochastic Hybrid Automaton (SHA) and showed that every transition is associated with an explicit event which is defined through a guard function. This also enables us to uniformly treat all events observed on the sample path of the SHS and makes it possible to develop a unifying matrix notation for IPA equations which eliminates the need for the case-by-case analysis based on event classes as in prior work involving IPA for SHS.

REFERENCES

L. M. Bujorianu and J. Lygeros. Toward a general theory of stochastic hybrid systems. In *Stochastic Hybrid Systems: Theory and Safety Critical Applications*, volume 337 of *Lect. Notes in Cont. and Inf. Sci.*, pages 3–30. Springer Verlag, Berlin, July 2006.

M.L. Bujorianu and J. Lygeros. General stochastic hybrid systems: modelling and optimal control. In *43rd IEEE Conf. on Dec. and Cont.*, volume 2, pages 1872 – 1877 Vol.2, dec. 2004.

C. G. Cassandras and Stephane Lafortune. *Introduction to Discrete Event Systems*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

C. G. Cassandras and J. Lygeros. *Stochastic Hybrid Systems*. Taylor and Francis, 2006.

C. G. Cassandras, Y. Wardi, B. Melamed, G. Sun, and C. G. Panayiotou. Perturbation analysis for on-line control and optimization of stochastic fluid models. *IEEE Trans. on Aut. Cont.*, AC-47(8):1234–1248, 2002.

C. G. Cassandras, Y. Wardi, C. G. Panayiotou, and C. Yao. Perturbation analysis and optimization of stochastic hybrid systems. *European J. of Cont.*, 16(6): 642–664, 2009.

M. K. Ghosh, A. Arapostathis, and S. I. Marcus. Optimal control of switching diffusions with application to flexible manufacturing systems. *SIAM*, 31(5):1183–1204, 1993.

M. K. Ghosh, A. Arapostathis, and S. I. Marcus. Ergodic control of switching diffusions. *SIAM*, 35(6):1952–1988, 1997.

P.W. Glynn. A gsmp formalism for discrete event systems. *Proc. of the IEEE*, 77(1):14 –23, jan 1989.

J. P. Hespanha. Stochastic hybrid systems: Application to communication networks. In *Hybrid Systems: Comput. and Cont.*, pages 387–401. Springer-Verlag, 2004.

X.D. Koutsoukos. Optimal control of stochastic hybrid systems based on locally consistent markov decision processes. In *Proc. of the IEEE Int'l Symp. on Intel. Cont.*, pages 435 –440, 2005.

S. Simić, K. Johansson, S. Sastry, and J. Lygeros. Towards a geometric theory of hybrid systems. In *Hybrid Systems: Comp. and Cont.*, volume 1790 of *Lect. Notes in Computer Sci.*, pages 421–436. Springer, 2000.

C. Yao and C. G. Cassandras. Perturbation analysis of stochastic hybrid systems and applications to resource contention games. *Frontiers of Electrical and Electronic Eng. in China*, 6(3):453–467, 2011.